DIGITAL FORENSICS

iOS forensics

September 7, 2021 by Hashim Shaikh

Share:

Day by day, smartphones and tablets are becoming ever more popular, and as a result, the technology used in development to add new features or improve the security of such devices is advancing too fast. iPhone and iPad are the game-changer products launched by Apple. Apple operating system (IOS) devices started growing popular in the mobile world. The latest smartphones or tablets can perform ideally most of the tasks which could be performed on a laptop or personal computer.

IOS devices provide larger storage space that could store emails, browsing histories, chat histories, Wi-Fi data and GPS data and more. From the forensics perspective, such devices could present lots of useful artifacts during the investigation. There are well-defined procedures to extract and analyze data from IOS devices which are included in this paper. This paper could be divided into the following sections. Introduction to the forensic processes focused towards mobile forensics, extracting logical and physical data from the IOS devices, IOS file system and storage analysis, analysis of logical data, data from the iTunes and iCloud back up, Wi-Fi and GPS data.

Overview of mobile forensics processes

In this Series

Mobile forensics is a field of digital forensics which is focused on mobile devices which are growing very fast. Due to the exponential growth of the mobile market, the importance of mobile forensics has also increased. The mobile phone generally belongs to a single person so analysis of it could reveal lots of personal information.

Due to the rapid growth, it also introduced challenges. The ratio of new models designed and launched is very high which makes it very difficult to follow similar procedures. Each case or investigation of the new model needs to be considered differently and requires following steps that could be different and unique to the case. With these challenges in mobile forensics, syncing mobiles phone to a computer using software becomes easy. One could extract data like SMS, contacts, installed applications, GPS data and emails, deleted data.

Collection

Below steps are recommended to follow during the collection of mobile device

- Note location from where mobile has been collected. It is good practice to take a picture using the camera of the location and mobile phone before starting any progress.
- Note the status of the device. Whether it's powered off or on. If it is power on then, check the battery status, network status. Check where the screen is locked.
- Search for the SIM package and if any cables are located around

Preservation

Preservation of evidence is a very crucial step in digital forensics. If it is very important to maintain evidence integrity throughout the investigation. For mobile forensics below steps are good practices to follow:

- It is possible that attackers could remotely wipe data or any new activity could override the existing data. So, the first step should be to isolate the mobile device from the network.
- There are several ways that could be followed according to the scenario,
- Removing SIM card
- Switching to Airplane mode
- Use Faraday's Bag or Jammer
- Chain of Custody – Chain of custody is the document to maintain each record of the Digital evidence from the collection to presentation. It includes details like serial no, case no, locker no,
- Investigator's name, time and date of each step, Details of evidence transportation. It is crucial because it keeps track of the Digital evidence.

iOS forensics

Top 7 tools for intelligence-gathering purposes

Kali Linux: Top 5 tools for digital forensics

Snort demo: Finding SolarWinds Sunburst indicators of compromise

Memory forensics demo: SolarWinds breach and Sunburst malware

Digital forensics careers: Public vs private sector?

Email forensics: desktop-based clients

What is a Honey Pot? [updated 2021]

Email forensics: Web-based clients

Related Bootcamps

Incident Response

- Hashing – Hashing is the method used to prove the integrity of the evidence. MD5 or SHA are widely used algorithms to calculate the Hash values of the evidence. As previously mentioned it is almost impossible to interact with mobile devices without altering them. But we could calculate the hash value of the extracted data through logical extraction or of the image file extracted through physical extraction.

Acquisition

There are three methods used for the data extraction from the IOS devices. Below overview has been given about each.

- Physical – It is a bit-to-bit copy of the device and allows recovering deleted data. Unfortunately, with mobile forensic always it is not possible to use this method.
- File system – This method would extract files that are visible at the file system level.
- Logical – This method allows to extract particular files from the file system like backup taken using iTunes
  Sometimes needs to perform offensive techniques like password cracking, Jail Breaking.

iOS devices and file system

Apple developed an operating system for iPhone, iPad and iPod Touch which is known as the IOS operating system. Devices running on IOS operating system are called IOS devices.

HFS+ file system

Apple developed Hierarchical File System (HFS) which provides large data sets. Disk formatted with HFS has 512-byte Blocks at Physical level.

There are two types of Blocks in the HFS.

Logical Blocks, which are numbered from first to last within the volume. They are also the size of 512 bytes same as physical blocks.

Allocation blocks are a group of logical blocks used to track data. Allocation blocks are further grouped together called clumps to reduce fragmentation on volume.

HFS uses both absolute time (Local time) as well as UNIX time so one can identify the location of the system.

HFS files system uses catalog file system to organize data. It uses B * tree (Balanced tree) structure to organize data. Trees are consisting of nodes. When data are added or deleted, it runs the algorithm to keep balance.

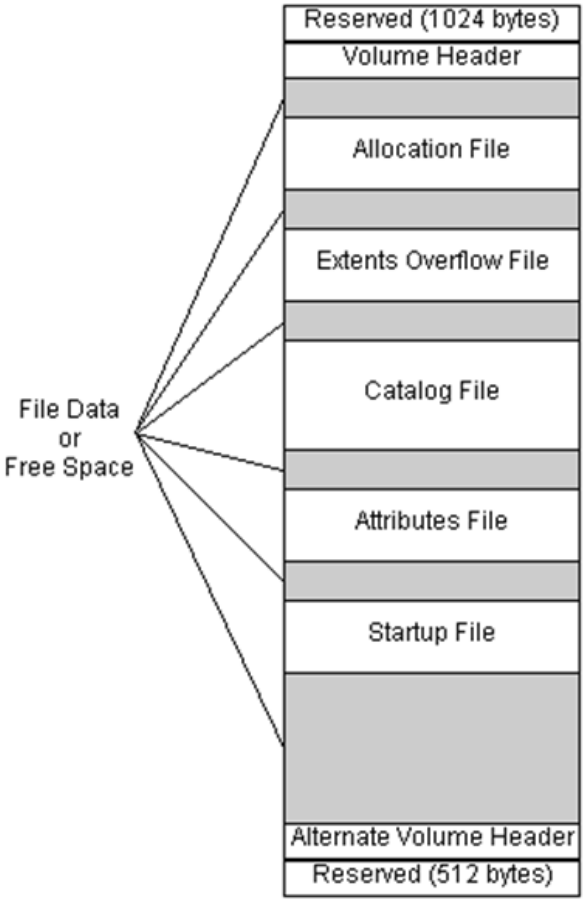


Figure 1. Structure of HFS+ File system

- As seen in above figure, first 1024 bytes are reserved boot blocks.
- Volume Header: This contains information about the structure of HFS Volume. It keeps track of Catalog ID Numbering and increases it one each time file added. HFS+ volume header also contains signature "H+."
- Allocation file: This keeps track of allocation blocks used by the file system. It basically includes a bitmap. Each bit represents the status of the allocation block. If it is set to 1, that means Allocation block is used, and if it is 0, that means allocation block is not used.
- Extent Overflow file: This consists of a pointer to the extent of the. If the file is larger than eight contiguous allocation blocks, then it uses extents.
- Catalog File: This organizes data using balanced tree system as mentioned previously. It utilizes to find the location of file or folder within the volume. It also contains the metadata of a file, including creation and modification date as well as permissions.
- Attribute File: This contains the customizable attributes of a file.
- Startup File: This assists the booting system which does not have built-in ROM support.
- Actual data is stored in the file system and tracked by the file system.

- Alternate Volume Header: This is a Backup Volume header located at the last 1024 bytes of the volume. It is 512 bytes long.
- The last 512 Bytes are reserved.
- HFSX File System

HFSX file system is a variation of HFS+ file system which is used in the Apple mobile devices. There is only one variation which is that it is case sensitive and it allows having two files with similar names but different case.

Partitions

IOS Devices have two types of partitions. System partition and Data Partition

System Partition

System partition does not contain more artifacts related to the investigation as it contains mostly system-related information like IOS operating system and pre-installed applications. The system partition is a Read-only as visible in below output of Private/etc./fstab.

Figure 1. fstab

An iPhone has a single disk, hence it is denoted as Disk0. The system partition is Disk0s1, and Data Partition is Disk0s2.

Figure 2. System Partition

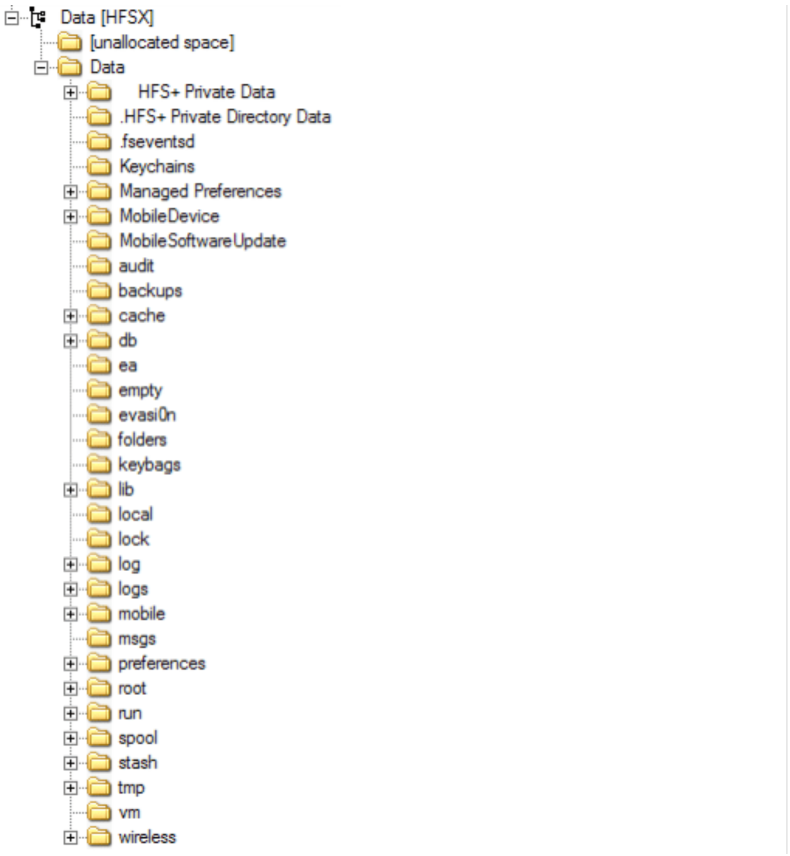
We can find the user-configured password from the /private/etc./passwd file as shown below.

Figure 3. Passwd file

As seen in above screenshot, mobile and root password hashes can be retrieved from the passwd file. Further using password cracking tool like “John the Ripper” one can get the password. The root password is “Alpine” and which is the default for all the IOS devices.

Data Partition

Data partition contains user data and can provide lots of artifacts during the investigation. It is a Read/Write partition. The structure of this partition has been changed with the different version of the IOS. Below is the screenshot from the IOS device which is running on IOS 7.



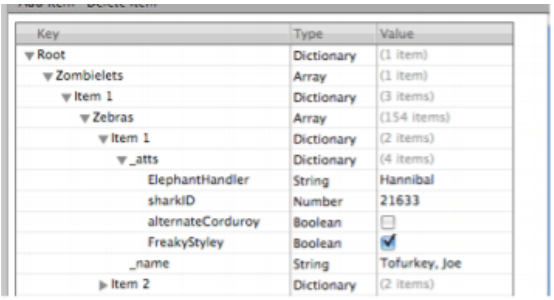
```
Data [HFSX]
    [unallocated space]
    Data
        HFS+ Private Data
        .HFS+ Private Directory Data
        .fseventsd
        Keychains
        Managed Preferences
        MobileDevice
        MobileSoftwareUpdate
        audit
        backups
        cache
        db
        ea
        empty
        evasi0n
        folders
        keybags
        lib
        local
        lock
        log
        logs
        mobile
        msgs
        preferences
        root
        run
        spool
        stash
        tmp
        vm
        wireless
```

Figure 4. Data Partition

The below directories are listed which could be of interest for the artifacts.

- Keychains – Keychain.db, which contains user password from various applications
- Logs – General.log: The OS version and Serial number, Lockdown.log – Lockdown Daemon log
- Mobile – User Data
- Preferences – system configurations
- Run – system logs
- Tmp -manifest.Plist: Plist Back up
- Root – Caches, Lockdown, and Preferences
- Property List Files

Property lists are the XML files used in the management of configuration of OS and applications. These files contain useful artifacts related to web cookies, email accounts, GPS Map routes and searches system configuration preferences, browsing history and bookmarks. These files could be open to the simple text editor to view the contents.



Key	Type	Value
▼Root	Dictionary	(1 item)
▼Zombielets	Array	(1 item)
▼Item 1	Dictionary	(3 items)
▼Zebras	Array	(154 items)
▼Item 1	Dictionary	(2 items)
▼_atts	Dictionary	(4 items)
ElephantHandler	String	Hannibal
sharkID	Number	21633
alternateCorduroy	Boolean	□
FreakyStyley	Boolean	☑
_name	String	Tofurkey, Joe
▶Item 2	Dictionary	(2 items)

Figure 5. Plist

SQLite databases

Logical extraction of the iPhone could provide lots of SQLite database files as it uses SQLite databases to store user data, the tool SQLite browser is used to explore and read SQLite database which can be download from http://sqlitebrowser.org/

The main three databases are Call History, Address Book, and SMS databases.

These databases could be extracted through applications available like SQLite database Browser as seen in the screenshot below.

Figure 6. SQLite Database Browser

Acquisition of iOS devices

Phone identification

During search and seizure, it is necessary that the examiner identifies the Phone model.

- One method is that check the back of the device which contains the model number printed

Figure 7. Model number printed on the back of the device

- Another approach is connecting iPhone to the forensic workstation. Install the library libimobiledevice on your workstation, it supports Windows, MAC and Linux up to 10.3 it can be downloaded from the URL http://www.libimobiledevice.org/ installation steps in details are explained here http://krypted.com/mac-os-x/use-libimobiledevice-to-view-ios-logs/
- Regardless of Phone is locked or unlocked; some information can be gathered about connected iDevice using command ideviceinfo as shown in below screenshot.

Figure 8. iDeviceinfo

As seen in the above figure, we could extract the following listed important information about iDevice

Device Class, Device Name, WiFiAddress, TelephonyCapability and HardwareModel, IOSversion

Operating modes of iOS devices

IOS devices can be operated in three modes. 1) Normal mode 2) Recovery mode and 3) DFU mode. It is necessary that examiner or Investigator should be aware of this mode as this knowledge is required to decide during the investigation that on which mode device should be operated to extract data or efficient extraction of data.

Normal mode

When iPhone is switched on, it boots in an operating system, this is normal mode. In normal mode, the user could perform all regular activities.

The normal mode boot process consists of three steps: Low-Level Bootloader, iBook and iOS kernel. These boot steps are signed to keep the integrity of the process.

Recovery Mode

The device enters into recovery mode if during the normal boot process if any step is failed to load or verify. The screenshot below shows the screen during recovery mode.

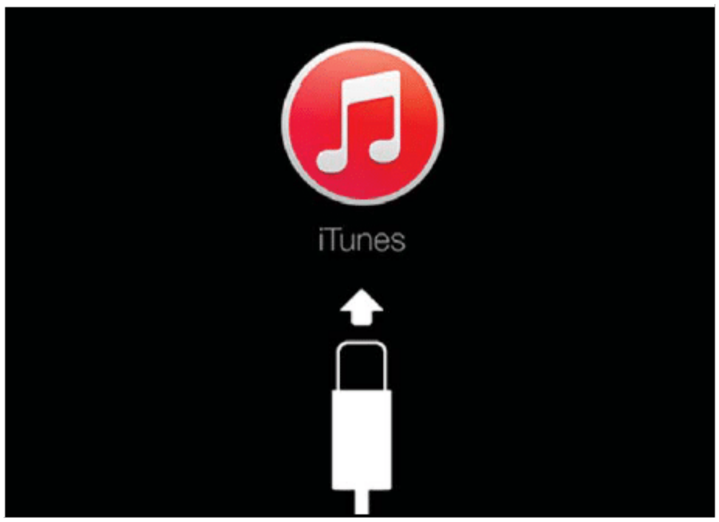


Figure 9. Screen during Recovery mode

This mode is used to perform upgrades or restore iPhone devices. iPhone can be entered in recovery mode by following the steps below:

- Turn off the device by holding the power button on the top of the device
- Hold home button of phone and connect it to the computer using a USB cable
- Keep holding home button till Connect to the iPhone screen doesn’t appear and then home button could be released.
- Reboot device to exit the recovery mode

DFU mode

Device Firmware Upgrade mode is used to perform IOS upgrading, and it is a low-level mode for diagnosis. During boot up, if Boot ROM is not getting a load or verify, then iPhone presents the Black screen.

The phone should be in DFU mode while using most acquisition techniques. Below steps needs to be performed to enter the iPhone in DFU mode.

- Install iTunes on a Forensic workstation and connect Phone to the forensic workstation using USB.
- Switch off Phone
- Hold the power button for 3 seconds
- Hold home button with power button hold for 10 seconds
- Release the power button and hold home button still didn't get alerted in iTunes that iPhone in recovery mode has been detected by iTunes.

Breaking passcodes

There are different methods of breaking the passcode of IOS. Depending on the version of IOS select the appropriate method. There are various tools that can perform such activity such as IP-Box, UFED lock recovery tool being a commercial tool and a python script in open source. We would be demonstrating a few of the methods of breaking passcodes for IOS.

Using IP-Box to break Phone passcode

If the device is locked using a four-digit passcode, then there are few tools available that could break this passcode.

The IP-BOX device does a similar task for more information you can visit the URL

IP-BOX is supported for devices up to IOS version 8.1.2. This kit contains Box, iPhone cable, USB cable, IP-BOX software used to configure patterns and using that IP-BOX firmware can be updated.

IP-BOX once connected to the iPhone as shown in below figure, it would send predefined passcodes to the phones. These codes are from 0000 to 9999. The screenshot below shows the detected passcode of the iPhone using IP-BOX

Figure 10. Passcode detected using IP-BOX

Using Python script to Bruteforce passcode

Performing Bruteforce attack on iPhone at springboard level could lead to wiping data within the phone. But this protection mechanism is not getting applied at kernel extension. Some tools can access the forensic workstation on which iPhone is connected and could perform brute force attack by accessing pairing key through an escrow file to decrypt phone.

To perform this examiner could follow the steps below:

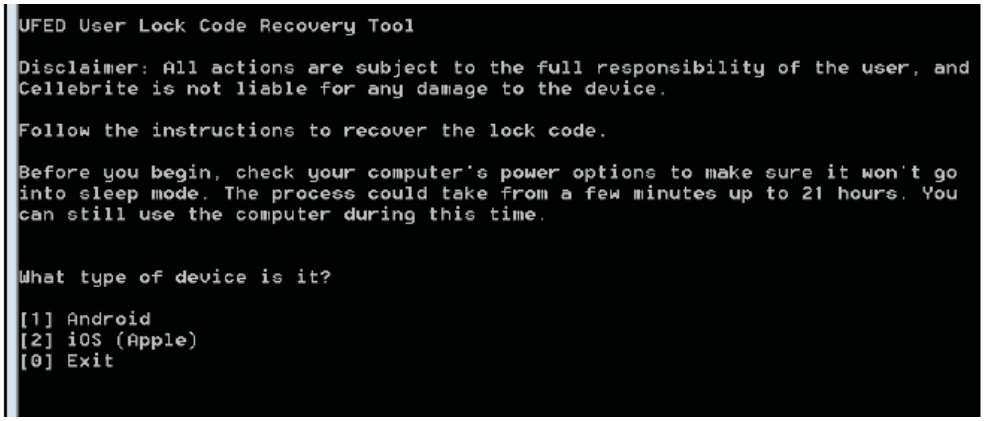
- Connect iPhone to the Mac system
- Get the script file from link and run the python script

These scripts communicate with the RAM disk on the Phone through Tcprelay.py with opened port 1999. This dumps the data protection keys into a directory named UDID by Brute forcing the system passcode and decrypting the System Keybag. Running the script would give the result as shown above. Now to brute force we need to hit the enter as mentioned by the script.

Figure 11. Bruteforce performed using Script

UFED User Lock Code Recovery

This is a commercial tool licensed under Cellebrite that uses the same technique as IP-BOX. It requires a cable and camera to sense the screen of the connected Phone. As per the screenshot below, it can crack passcodes of IOS devices as well as Android.



```
UFED User Lock Code Recovery Tool

Disclaimer: All actions are subject to the full responsibility of the user, and
Cellebrite is not liable for any damage to the device.

Follow the instructions to recover the lock code.

Before you begin, check your computer's power options to make sure it won't go
into sleep mode. The process could take from a few minutes up to 21 hours. You
can still use the computer during this time.


What type of device is it?

[1] Android
[2] iOS (Apple)
[0] Exit
```

Figure 12. UFED User Lock Code Recovery Tool

Direct acquisition

iDevice browser can be used to directly acquire data if the phone is not locked or lock down certificates is known. This is a very simple method as upon connecting the phone to the forensic workstation, iDevice Browser lists the files as shown below.

Figure 13. iDevice browser

Such software is working on forensic platforms that mean they could modify the data or accidentally override the evidence. iMazing, iFunBox, iExplorer, Wondershare Dr. Fone are tools that use the libraries from iTunes hence it requires an updated version of iTunes. These tools are running on Windows or MAC platforms. Before connecting the iPhone to the device, make sure the automatic syncing option is enabled on iTunes.

This method is a very simple way to copy data using the browser. One can use logical acquisition method using the iDevice browser as an alternate, and it depends on the scenario. Another tool that can perform logical acquisition is described below.

Logical acquisition

Logical acquisition can be performed with the help of various commercial tools such as Oxygen Forensic Suite, UFED physical analyzer, Cellebrite, Blacklight, XRY. We would be demonstrating Logical Acquisition with the help of Oxygen forensic suite and UFED physical analyzer tools below.

Logical Acquisition using Oxygen Forensic Suite

Using Oxygen Forensic Suite which is a commercial tool, we could perform logical acquisition of iPhone. Steps as described as below

- Launch the Oxygen Forensic Suite. Select the Connect device option to start extraction.
- It would prompt to select automatically connect a device or manually connect the device. It is recommended to use the First option which is automatically

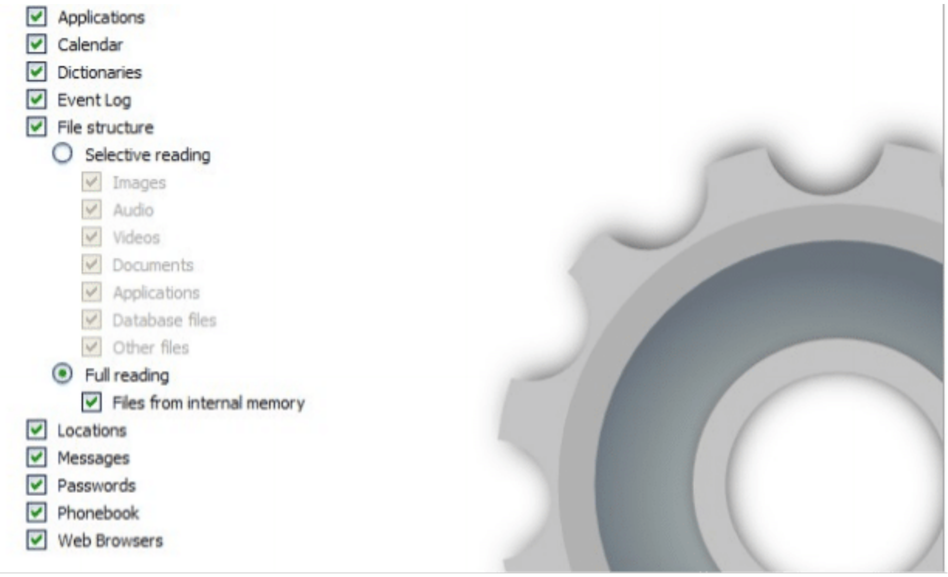
connecting the phone. Oxygen forensic suite would start searching for the phone once selected automatically connects option.

- The software will provide UUID of the detected phone and if the phone is password protected and locked it would ask to provide a password or lock down certificate.

As shown in below figure, if the password is known then examiner needs to enter and authorize password on the device and select the option “I entered the passcode. Press to connect” or select lockdown plist.

Figure 14. Enter a passcode or choose lockdown certificate

- After the successful connection gets established, the software would display information about the connected device. Information is like a model, IMEI number, boot loader and IOS version information.
- The next window provides an option to insert case-related data like Device name, Device Owner, Evidence Number. It also asks password for the Backup.
- Next windows would allow choosing types of data that want to be extracted. Recommended action is to select all.



- ☑ Applications
- ☑ Calendar
- ☑ Dictionaries
- ☑ Event Log
- ☑ File structure
  - ○ Selective reading
    - ☑ Images
    - ☑ Audio
    - ☑ Videos
    - ☑ Documents
    - ☑ Applications
    - ☑ Database files
    - ☑ Other files
  - ● Full reading
    - ☑ Files from internal memory
- ☑ Locations
- ☑ Messages
- ☑ Passwords
- ☑ Phonebook
- ☑ Web Browsers

Figure 15. Data to be extracted

- Software now starts to extract data, and at the same time, it parses the data extracted. If Phone backup is password protected, then the extractor would pass data to the Passware kit to perform the attack.
- If examiner knows the backup password, then password cracking step could be skipped by the examiner and could supply the password. If password cracking is successful then it would extract all the data from the Backup otherwise only multimedia data like Images, Videos would be extracted, and examiner would not be able to get any information about installed or preinstalled applications.

Figure 16. Passware Kit password cracking

Advance logical acquisition or file system dump

A file system dump or File system acquisition is usually referred to as an advanced logical acquisition, which is a subset of a physical image, could be performed by several well-known tools such as Cellebrite, Blacklight, Oxygen or XRY. As it provides access to the file system data, it allows more data to be extracted than logical acquisition. We would perform a file system dump using UFED.

Using UFED Physical Analyzer to perform Advance logical acquisition

To perform advanced logical extraction using UFED physical analyzer, the first step is to select the option to start IOS Device Extraction from the Extract menu of the Main Window.

Figure 17. IOS device Extraction

The next option would allow choosing Advance logical extraction or Physical mode. Once we select advanced logical extraction, it is necessary to have the phone connected to the Forensic Workstation as shown in the screenshot below.

Figure 18. Advanced logical extraction

The above figure shows which kind of artifacts tool could extract using Advance logical Extraction.

If the device is locked, it would through an error message “iOS device is locked.”

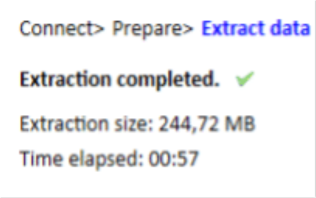
The analyst needs to unlock the device or put the lockdown certificate on the correct directory. While extracting the backup if the password is set, there are two methods offered by the tool.

Method 1: provide the password of the backup or crack the backup password and extract the full data

Method 2: extract partial data which are available to extract without cracking back up

Figure 19. Extraction Method

Upon selecting the extraction method, the Software would allow the user to choose a destination folder to extract the data. Method 1 is faster than method 2. Once extraction finishes, the tool shows a report stating the size of the data extracted and time taken.



Connect> Prepare> **Extract data**

Extraction completed. ✓

Extraction size: 244,72 MB
Time elapsed: 00:57

Figure 20. Extraction Status

Physical acquisition

Using physical acquisition, the examiner could extract almost all data by accessing phone memory and all files stored there. iOS devices use two types of memory: volatile and non-volatile. RAM load executes important parts of the operating system or application. It gets flushed once the device reboots. Username, passwords, encryption keys and more important artifacts could be found from the RAM. From a forensic perspective, it is crucial to extract information stored in RAM.

NAND (non -volatile) memory would keep the data if it rebooted. System files and user data are stored in NAND flash. Using physical acquisition, bit by bit copy of the NAND can be acquired.

Using custom RAM disk for the physical acquisition

This method uses weaknesses in the boot process while the device is in DFU mode to load a custom RAM disk and get access to the file system. Forensic tools in the custom RAM disk dumps the file system over USB via SSH tunnel. If it is performed properly, then data within the phone would remain intact, and there will not be any alteration to data. So, there is less risk of distortion of evidence data.

iPhone’s secure boot chain prevents loading RAM disk. Hence we use the method to exploit the BootROM vulnerability and patching stages as shown in the figure below.

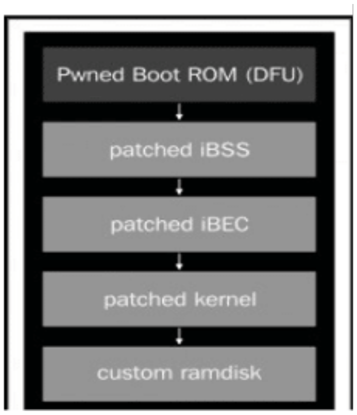


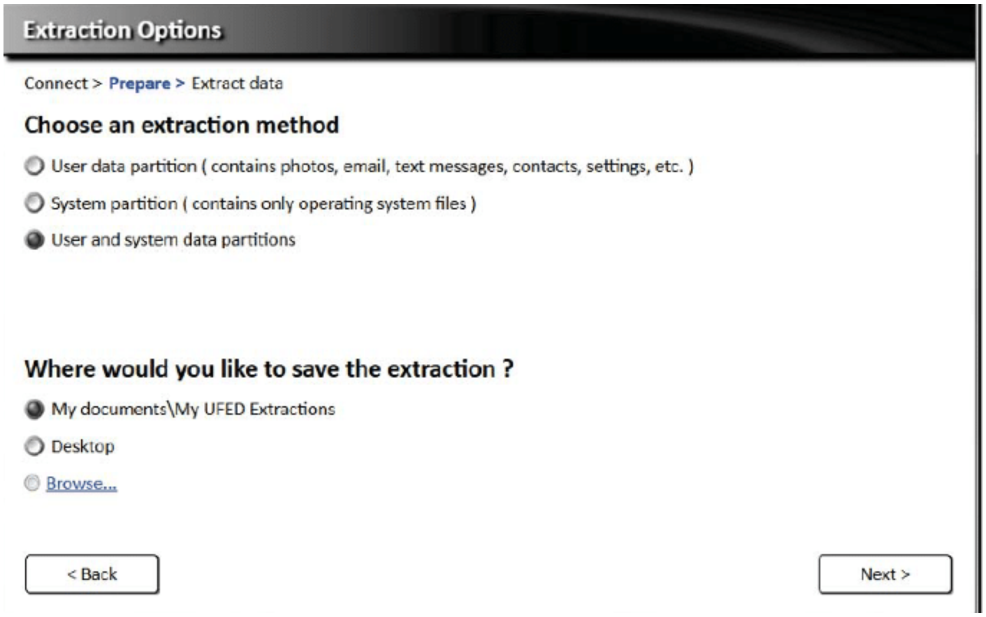
Figure 21. Custom RAM disk

Forensic tools leverage this vulnerability to perform physical acquisition of iOS devices. The steps below show the process to acquire physical access to the IOS device using the UFED physical analyzer. This tool instructs the examiner to switch the device into DFU mode. Once the device is placed in the DFU mode, the tool will verify and provide an alert indicating that it is possible to perform physical acquisition using this method for this device or of the need to perform another method of acquisition.

Once we proceed with the physical acquisition, the tool will start uploading Bootloader to iPhone.

As soon as Bootloader is uploaded to the phone, the tool gets access to the NAND flash. It prompts the screen to extract system data or user data. If the connected phone is not jailbroken, then the system partition would be in read-only mode.

As shown in the figure below, to acquire a full image, both data and system partition need to be selected. Most of the tools dump data and system partition as a single disk image. The selected tool dumps the image to a destination folder as shown below:



Extraction Options

Connect > Prepare > Extract data

Choose an extraction method

○ User data partition (contains photos, email, text messages, contacts, settings, etc.)

○ System partition (contains only operating system files)

● User and system data partitions

Where would you like to save the extraction ?

● My documents\My UFED Extractions

○ Desktop

○ Browse...

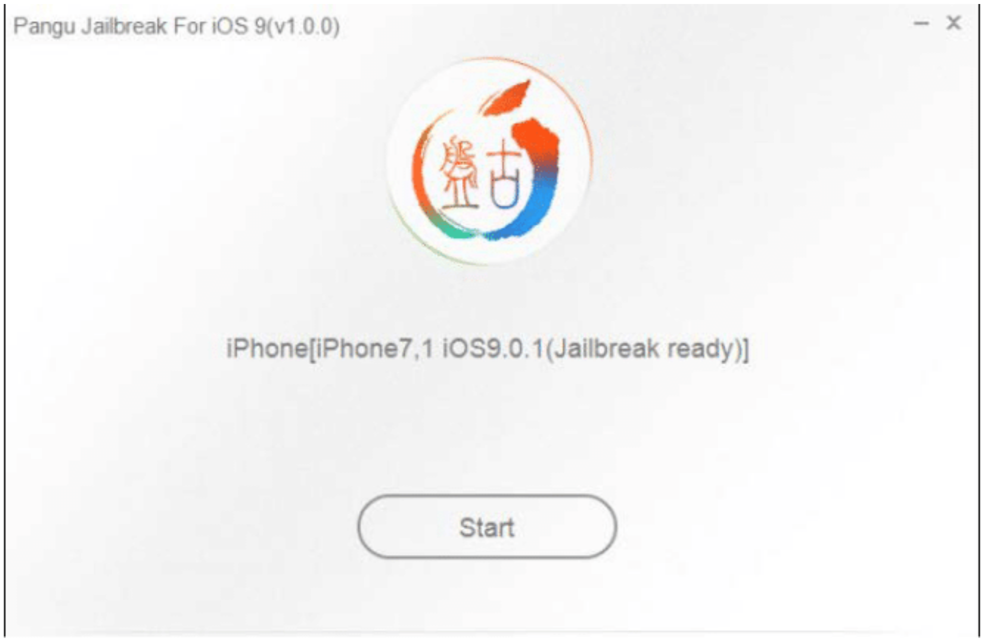
Figure 22. Dumping Image of both partitions to the destination

Jailbreaking an IOS device

This covers a method to Jailbreak IOS device 9. Using this method user could gain access to all partitions including the file system with read-write. This method is not accepted forensically because it may overwrite some important data but it allows performing physical acquisition efficiently. Jailbreak is a method to remove software restrictions and expand the feature set by a manufacturer like Apple in the case of IOS. This method is used when root access of the file system is required as well as for the physical acquisition.

In this example, we used the tool Pangu Jailbreak, which can be found at http://www.downloadpangu.org/. Steps are described below,

- Download the latest version of Pangu Jailbreak, using a USB connection to connect the phone with PC.
- Make sure iTunes is closed or not running.
- Disable passcode and switch iPhone to Airplane mode.
- Launch application Pangu Jailbreak.
- As shown in figure 23, select the start button once the Device is detected by the application Pangu Jailbreak.



iPhone[iPhone7,1 iOS9.0.1(Jailbreak ready)]

Figure 23. Pangu Jailbreak Launch

- Proceed further with option Already Backup as shown below

Figure 24. Jailbreak notice

The figure above shows the notice presented by the application. This warns the user about Data loss may occur as well as for the smoother and successive operation suggests switching the phone to airplane mode. It also suggests backing up data before proceeding further.

- Once the user selects Already Backup, the Tool starts the process and the status of progress is indicated in percentage form in the Application Window. At 55% progress, the device may reboot and at the progress at 65% asks to re-enable Airplane mode.

As shown in the screenshot below, at the progress of 75%, would be asked to unlock the device and run Pangu Jailbreak.

Figure 25. Status at 75% progress

- The application will now ask to allow Photo access permission for some unknown reason. Upon Finishing, the Phone reboots and Pangu prompts that the device is already Jailbroken,

Figure 26. Device is Jailbroken

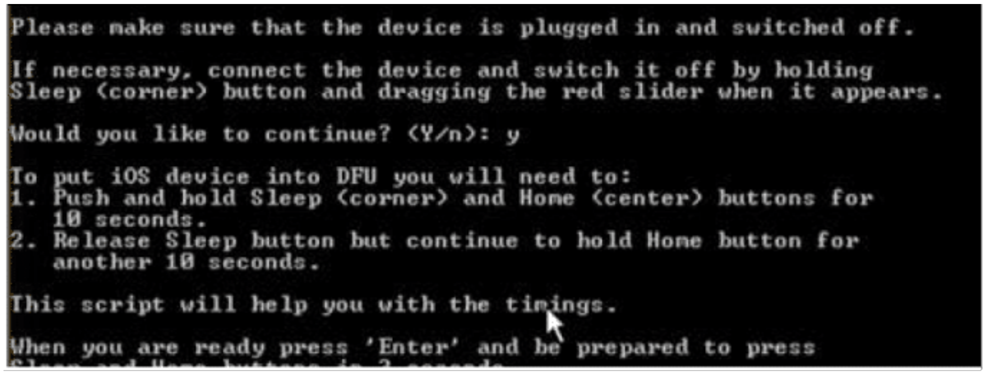
Physically acquire IOS device

A tool such as XRY, Cellebrite, Lantern, MPE, Elcomsoft could be used for the physical acquisition of IOS devices. The output of all the mentioned tools would either be a bitstream image (dd) or a DMG image file that can then be analyzed manually or with the help of a forensic analysis tool. We'll demonstrate physical acquisition using Elcomsoft.

Using Elcomsoft IOS Forensic Toolkit to Physically acquire IOS device

Elcomsoft IOS Forensic Toolkit is a commercial tool that allows us to take a Bit to Bit Image of iOS devices. It also supports the extraction of secret passwords and decryption of file systems.

- Turn off the IOS device and connect it to the Forensic workstation or PC. Now select Option 1 from the Wizard to make the phone enter DFU mode.

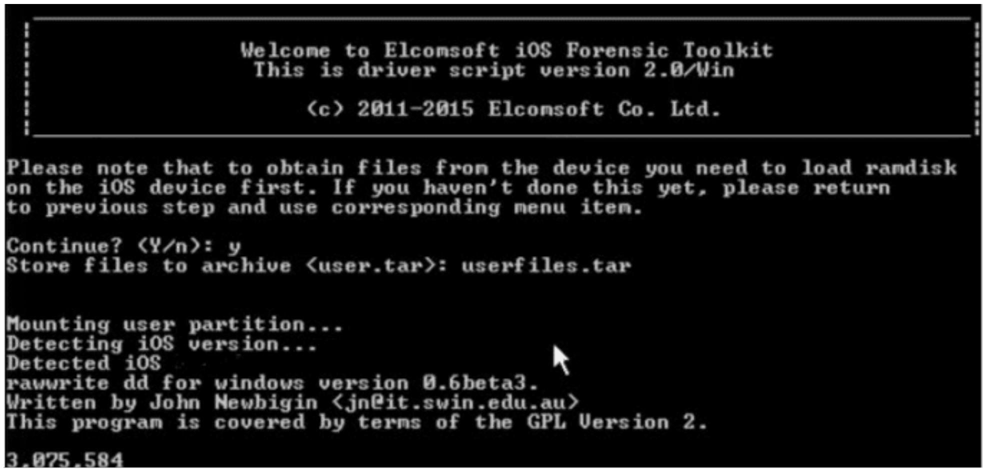


```
Please make sure that the device is plugged in and switched off.

If necessary, connect the device and switch it off by holding
Sleep (corner) button and dragging the red slider when it appears.

Would you like to continue? (Y/n): y

To put iOS device into DFU you will need to:
1. Push and hold Sleep (corner) and Home (center) buttons for
   10 seconds.
2. Release Sleep button but continue to hold Home button for
   another 10 seconds.

This script will help you with the timings.

When you are ready press 'Enter' and be prepared to press
```

- After this step is finished, the tool will ask to allow to upload a custom RAM disk

Many important artifacts like Apple ID/Password, Wi-Fi passwords, and VPN credentials can be extracted from keys.plist

All user data can be extracted using option 8, which acquires all user's files as a tarball. This process takes time depending on the amount of user data. Upon finishing, the data extracts the .tar file. The figure below shows this process.



```
 ________________________________________________________________________
|                                                                        |
|                 Welcome to Elcomsoft iOS Forensic Toolkit               |
|                   This is driver script version 2.0/Win                 |
|                                                                        |
|                      (c) 2011-2015 Elcomsoft Co. Ltd.                  |
|________________________________________________________________________|

Please note that to obtain files from the device you need to load ramdisk
on the iOS device first. If you haven't done this yet, please return
to previous step and use corresponding menu item.

Continue? (Y/n): y
Store files to archive <user.tar>: userfiles.tar


Mounting user partition...
Detecting iOS version...
Detected iOS
rawwrite dd for windows version 0.6beta3.
Written by John Newbigin <jn@it.swin.edu.au>
This program is covered by terms of the GPL Version 2.

3.075.584
```

Figure 28. User’s file

Using option 6, the User can acquire a physical image of the device file system. The tool would prompt to select the option to choose whether to extract system and user data. Please refer to the screenshot below, in which option 2 has been chosen. Encrypted user data will then be extracted. To decrypt this data, we can use keys.the plist file which we extracted in the previous step.

Figure 29. Physical Acquisition Elcomsoft

Figure 30. Decrypting user data

Using this decryption, we'll get the image file user file-decypted.dmg, which is ready to mount.

Analysis

Data structure and artifacts

This section covers the analysis of the important artifacts that are generated by features of the system or interaction of the user with the device.

- It is necessary to understand how data is stored in the device. Most of the user data is stored under /private/var/mobile/ or /User/ which is symlink pointing to previously mentioned directory.

/private/var/mobile/Application: /User/Application points to this actual path

/User/Applications/ ######-####-####-####-###########: # represents the UUID

<Application_Home>/AppName.app: This file contains application bundle. This file doesn't get backed up

<Application_Home>/Documents/: This folder contains application related data files.

<Application_Home>/Library/: This also holds application specific files.

<Application_Home>/Library/Preferences/: This directory contains application preference files.

<Application_Home>/Library/Caches/: This folder holds Application specific support file and doesn't get backed up.

<Application_Home>/tmp/: This folder contains temporary files.

- iTunesMetadata.plist from the root application folder contains information related to device, Apple account name, date of purchase. This information could be useful during investigation in some cases. In each application directory one of these files would be found.

The figure below shows the file and detail found from the file.

Figure 31. iTunesMetadata.plist

Mostly two types of files are encountered in the apple device:

Plist: used for configuration files, mainly

SQLite databases

Timestamps

While analyzing artifacts, it is important to determine the timestamp of that artifact. IOS device uses MAC Absolute time. There are resources available that can convert this timestamp to human-readable time. This is also performed using the date command with u switch on Mac, which will display time in UTC or on local time.

Databases

SQLite databases are most common data storage in IOS devices as well as other mobile platforms like Windows Phone. These databases are used to store data of native as well as third- party applications.

There are many free/open-source tools available to explore SQLite database files. The most popular and widely used application is SQLite Database Browser, which has both GUI and command-line utility.

Property List files

Plist or Property List files are the most commonly used data formats in IOS devices. These files stores configuration information, preferences, and settings.

These files could be explored using a simple text editor. A tool commonly used to parse these files is plist Editor.

Configuration files

Information extracted from the configuration or preferences files could be important which investigation. Some important configuration files are listed below

- Account and device information – /private/var/root/Library/Lockdown/data_ark.plist contains device and its account holder's information.
- Account information – Path of file is as below

/private/var/mobile/Library/Accounts/Accounts3.sqlite . This file holds account information.

/private/var/mobile/Library/ DataAccess/AccountInformation.plist – This file contains information of the account which used to set up applications.

- Airplane Mode – File available at below mentioned path shows whether airplane mode is enabled or disabled currently on the device.
- /private/var/root/Library/Preferences/com.apple.preferences.network.plist.
- Installed application list – /private/var/mobile/Library/Caches/com.apple.mobile.installation.plist
- Above mentioned file contains all installed application's list with paths of each application's files. This is very useful while mapping GUIDs to specific applications.
- AppStore settings – Last search store could be find from the below mentioned file.
- /private/var/mobile/Library/Preferences/com.apple.AppStore.plist
- Configuration information and settings – /private/var/mobile/Library/preferences/
- Apple application's settings and configurations could be extracted from the plist files on mentioned folder.
- Lockdown certificate information – computers paired with iOS devices and lockdown/pairing certificates are contained in below path
- /private/var/root/Library/Lockdown/Pair_records/
- Network Information – /private/ var/preferences/Systemconfiguration/com.apple.network.identification.plist – this plist file contains ip networking information cache like router, network addressed and server used previously. It also provides the timestamps.
- Notification log – /private/var/mobile/Library/BullitenBoard/ClearedSections.plist – Log of cleared

notifications are contained within this plist file.

- Passwords – The password saved in iOS 10/9/8/7 are found form file from path /private/var/keychains/Keychain-2.db
- SIM card info – /private/var/wireless/Library/Preferences/com.apple.commcenter.plist – this plist files holds ICCID and IMSI of the SIM card last used.
- Springboard – This plist file contains order of applications in each screen. Path for this file is
- /private/var/mobile/Library/Preferences/com.apple.springboard.plist
- System Logs – iOS system logs are available in below path
- /private/var/logs/
- Wi-Fi networks – /private/var/preferences/ SystemConfiguration/com.apple.wifi.plist -Known Wi – Fi networks, timestamp of last joined and important information can be gathered from this file.

Preinstalled IOS applications

IOS devices are provided with some pre-installed applications like Safari browser, e-mail client, calendar and basic phone function utilities like Camera, Call history. These artifacts can be found in the application folder itself.

Data related to the communication, preferences, Internet history and cache, keyboard keystrokes can be found from the Library folder,

/private/var/mobile/Library/ – If data is acquired through Physical acquisition, this path has files mentioned above.

Backup service/mobile/Library/ – path for File system acquisition

Library – Path from the Logical acquisition

/private/var/mobile/Media/ or Media folder is also the important location that can provide artifacts like audio and pictures created by the user.

Address Book

Contacts, Application related to personal contacts in SQLite database file format are available in Library folder of AddressBook folder. Two important databases are available in this folder one is AddressBook.sqlitedb and second AddressBookImages.sqlitedb.

AddressBook.sqlitedb database file provides details like name, surname, email address and phone number of each contact. This information is saved in tables. The main tables are ABPerson and ABMultiValue.

AddressBookImages.sqlitedb – Images associated with a contact are stored in this database. These images appear on the screen when that contact calls. ABFullSizeImage is the table name which stores these images in Database.

Audio recordings

A preinstalled app like Voice memo allows the user to record voice memos. These memos are stored at location /private/var/mobile/Media/Recordings/.

In folder mentioned above file named Recordings.db holds information about each voice memo. Information like date, duration, memo name and filename of an audio file could be extracted from this database file.

Calendar

Manually events can be created within the calendar as well as it syncs events with other applications. This information is stored in two database files

/private/var/mobile/Library/Calendar/Calendar.sqlitedb – This database file contains information related to events available in calendar

/private/var/mobile/Library/Calendar/Extras.db – This database file holds information like calendar settings or details related to notification to particular calendar event.

Call History

Call Details like incoming calls, Outgoing calls and missed calls can be extracted from the /private/var/wireless/Library/CallHistory/Call_History.db. It also provides the date, time, and duration of the call.

IOS 8 has different path for this file which is as mentioned below

/private/var/wireless/Library/CallHistoryDB/CallHistory.storedata

DialerSavedNumber – Last dialed phone can be extracted from the plist file from the below location

/private/var/mobile/Library/Preferences/com.apple.mobilephone.plist

This file remains though user deletes call history database file so forensically this is a very important file.

Another important file with the forensic perspective is com.apple.mobilephone.speeddial.plist – contains speed dial numbers. Location for this file is as below

/private/var/mobile/Library/Preferences/com.apple.mobilephone.speeddial.plist

Email

Mails sent, received, or drafted are stored in the database at below mentioned path

/private/var/mobile/Library/Mail/. Each account has a separate folder within the Apple Mail application.

Images

/private/var/mobile/Media/ contains photos inside IOS devices. There are two folders

DCIM – contains user created photos like captured by camera or screenshots.

PhotoData – This folder contains photo albums synced with clouds as well as a computer.

Thumbnails for each image are stored in /Private/var/mobile/Media/PhotoData/Thumbnails/

Photos.sqlite database contains information about images.

Using Thumbnails and information about images can be recovered irrespective of the original image is available or deleted.

Maps

IOS devices have Apple's own MAPS application. Information related to the last searches, like search query or Longitude and Latitude coordinates can be retrieved from below path

/private/var/mobile/Library/Preferences/com.apple.Maps.plist.

The main folder of Maps contains information of the searches of users and bookmarked locations. It is located at path

/private/Library/Maps

Notes

User created notes are stored at /private/var/mobile/Library/Notes/notes.sqlite.

The ZNOTE and ZNOTE-BODY tables contain important details like note title, content, creation, and modification date.

Safari

Every IOS device has the Safari browser preinstalled. There are two locations where all activities get stored. These locations are /private/var/mobile/Library/ and main application folder of Safari.

Safari Bookmarks – Saved bookmarks in Database file are found from /Library/Safari/Bookmarks.db

Last time bookmarks were modified timestamp can be extracted from the plist file /Library/Safari/Bookmarks.plist.anchor.plist

Safari Cookies – Web sites cookies are stored in /Library/Cookies/Cookies.binarycookies

Safari Screenshots – Thumbnails of web pages visited can be found from the directory Library/Caches/Safari/

Search cache – Most recent searches entered into the search bar can be retrieved from plist file at location Library/Caches/Safari/Recentsearches.plist

Search History – Library/Preferences/com.apple.mobilesafari.plist contains recent search list. This file is important as a forensic perspective because when the user would delete cache or history from the browser then as well this file would not be deleted.

Suspended state – Library/Safari/Suspendedstate.plist. This plist file contained state of safari when user powered off iPhone or browser got crashed. This file would contain a list of URLs open at the time of state occurred.

Safari Thumbnails – /Library/Safari/Thumbnails/ – screenshot of the last active browser pages viewed by third party apps are contained in this folder.

Safari Web Cache – Library/Caches/com.apple.mobilesafari/Cache.db contains recently downloaded and cached objects in safari.

Safari History – Library/Safari/History.plist file contains web browser history. If history is cleared by the user then it this file would not contain history.

SMS

A database which stores SMS, MMS and iMessages sent or received at /private/var/mobile/Library/SMS/sms.db

Attachments in the SMS or MMS or iMessage are stored at /Library/SMS/Attachments/

Drafts are saved at /Library/SMS/Drafts. Each draft has its own plist file within this folder.

Voicemail

The Voicemail folder at /private/var/mobile/Library contains AMR codec audio files of each voicemail recorded message and voicemail.db database, where information related to each audio file like sender, the date, the duration and so on are stored.

General IOS forensics artifacts

Artifacts covered in this section are not related to any particular application, but those are the evidence generated by general use of the IOS device.

Clipboard

Cached copy of device’s clipboard data can be found in a binary file located at /private/var/mobile/Library/caches/com.apple.UIKit.pboard

Data like passwords or other portions of text data which copied, cut, or pasted by the user are stored in this file.

Keyboard

Auto correction and auto-completion while typing are supported by IOS devices. Device caches user types in file dynamic-text.dat file. A file can be found via path /private/var/mobile/Library/Keyboard. In the same directory IOS stores, one file for each language is used and configured on the keyboard.

Location

The consolidated.db file which is Consolidated GPS cache contains location information related to every Wi-Fi hotspot and cell tower that phone ranged. This database is located at /private/var/root/Library/Caches/locations/consolidated.db. Table WiFiLocation and CellLocation are important in this database file.

In newer devices, a new database has been included instead of consolidated.db which are /private/var/root/Library/Caches/locations/cache_encryptedA.db – similarly like consolidated.db this database contains Wi-Fi access points and cell towers that ranged with the device. The main difference is data remains only for 8 days before being cleared out.

Other applications which track the geographical locations may store GPS coordinates and timestamps.

Snapshots

Fade-out effect is being used by IOS for the transition of two screens. As a part of this IOS saves screenshots of the current screen and fade out effect applied picture of the current screen.

These screenshots are stored at below locations

/private/var/mobile/Library/Caches/Snapshots/ contains apple preinstalled applications

/private/var/mobile/Applications/<app_UUID>/Library/Caches/Snapshots – contains all application snapshots. Using these feature lots of forensically valuable information can be gathered. There is a possibility that screenshots of SMS are available, but that SMS is no longer available because deleted.

Spotlight

This feature is to assist the user to assist while searching like applications, SMS, contacts and more.

/private/var/mobile/Library/Spotlight/ stores spotlight indexes and searches.

In the directory mentioned above, there are two folders related to SMS searches and another of general spotlight utility.

Wallpaper

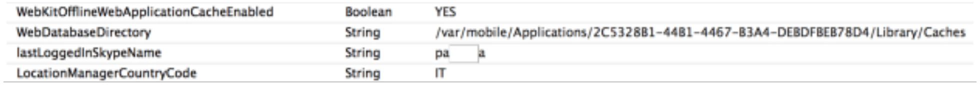
/private/var/mobile/Library/Springboard/ stores images used as wallpaper. Two types of wallpaper images are available. HomeBackgroundThumbnail.jpg which is related to the wallpaper when the device is unlocked. LockBackgroundThumbnail.jpg is related to the wallpaper when the device is locked.

Third-party applications

This section covers third-party application analysis which includes locations of the important artifacts related to third party applications.

Skype: This is most popular VoIP and chat application

com.skype.skype.plist file inside preferences folder contains skype username as shown in below screenshot.



WebKitOfflineWebApplicationCacheEnabled	Boolean	YES
WebDatabaseDirectory	String	/var/mobile/Applications/2C5328B1-44B1-4467-B3A4-DEBDFBEB78D4/Library/Caches
lastLoggedInSkypeName	String	pa[redacted]a
LocationManagerCountryCode	String	IT

Figure 32. com.skype.skype.plist

The above screenshot shows only the username which last logged on. Other folder Library/Application Support/Skype/ contains all profiles logged in from this device. This folder contains one folder for each account logged in with that device as shown in below figure below. Each user folder has the database which contains information like chats, contacts. main.db file contains all information stored in clear.

There are utilities available which parse these files. In the application folder voicemail messages, screenshots can be found.

Figure 33. Skype user profile

WhatsApp

WhatsApp is widely used instead of SMS nowadays, as it is an Instant messaging application.

net. whatsapp.whatsApp.plist file is located at Library/Preferences/. This plist configuration file provides basic information like username, phone number to which WhatsApp account is associated and more.

Table ZWAMESSAGE contains exchanged messages, timestamps and user's name who was chatting which is visible in below screenshot.

Figure 34. ZWAMESSAGE

Open single user or group chats are stored in table ZWACHATSESSION. Information in this table could be correlated with the information in table ZWAGROUPMEMBER and ZWAGROUPINFO. References to the multimedia files exchanged, timestamps and location of multimedia file are stored in table ZWAMEDIAITEMS.

Documents/ChatSearch.sqlite – in this database file table docs_content contains chat contents.

Facebook

It is most widely used social network application. Because of this, during the investigation, high amount of information from the Facebook application can be found. Mainly three kinds of information can be found – user personal information, caches of visited profiles and pages, information about external sites visited within the Facebook application.

Account information can be found from the Library/Preferences/com.facebook.plist file. Email address and Facebook id of configured account on the application could be extracted from this file. The date of the last time application was also used could be retrieved from this file.

Contact related information are stored in Library/Caches/FbStore.db

Profile pictures are saved in Library/Caches/ImageCache/ Directory.

Images viewed while surfing through pages of social network can be found from below folder

/Library/Caches/_store_<APP_ID>/<ios-Version>_<language>/FBDiskCache/

Library/Caches/com.facebook.Facebook/Cache.db and Library/Caches/com.facebook.facebook/fsCacheddata/ – contains contents of the other website visited including related URL and corresponding files.

Videos watched by user within this application are stored in Library/Caches/com.facebook.facebook/var/mobile/Applications/../video_url_cache/Cache.db.

Cloud storage applications

As the cloud allows the accessing of stored data from anywhere and anytime, Cloud storage applications have become very popular. It is very frequent that this kind of application would be encountered during the investigation. Analysis of few popular cloud storage applications is covered in this section.

Dropbox

This application is saved in /private/var/mobile/Applications/4BD80D3B-7ADA-4171-B2A0-8A534F05408D/

There are four subfolders in this folder: Cookies, DropboxPrivate, preferences, and Cache

Local copies of opened files are saved on Cache folder. It could acquire only through physical acquisition.

A file named com.getdropbox.Dropbox.plist is stored in the preferences folder. This plist file contains user information like name, surname, and email. The application structure is shown in the screenshot below:

Figure 35. Dropbox application structure

Google Drive

/private/var/mobile/Applications/8F139264-9142-4B84-A7C3-421ADD6BA05F/ is the path where Google drive application is stored. There are two subfolders within it. Documents and Library. In Library another subfolders Caches, preferences, and Cookies.

User information like name and surname, User ID and User e-mail can be found from plist file com.google.Drive.plist exists in the preferences folder.

Cached copies of opened files are stored in Caches. It could be extracted if the dump is acquired using Physical acquisition.

Three databases followed Contacts_snapshot_useremail.db, Feed_snapshot_usermail.db and Items_snapshit_usermail.db is stored inside Documents folder.

User’s email ID, name, and shared files are stored in contacts db. All information listed below about files stored is in Items db.

Identifier, Title, Kind, MD5 hash, Last Modified By, Last Modified Date, Last Viewed Date, Shared with Me Date, Last Modified by Me Date

Figure 36. Google drive application structure and plist file

File Carving

Data protection technology is used by Apple in flash memory on IOS devices. 256-bit per file key generated every time a new file is created, and content of the file is encrypted using AES encryption. This per-file key is later wrapped with class key and stored in the file's metadata, which is then encrypted with the EMF (file system) key. EMF key is generated by unique hardware UID. This process is shown in the figure below.

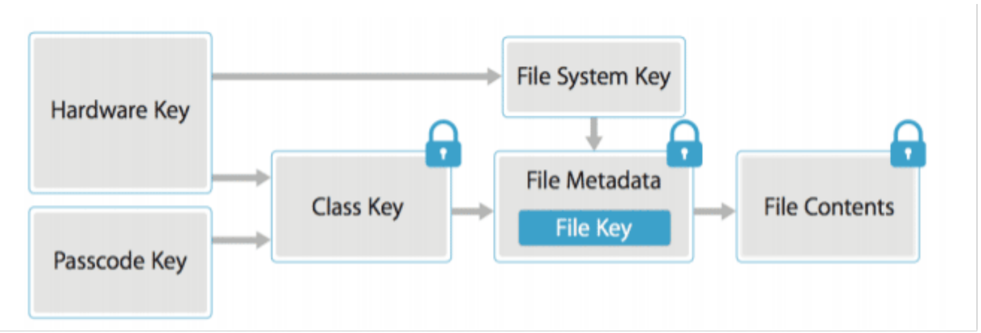


Figure 37. Data Protection

Due to this data protection schema, it is not possible to carve files with the classical approach. By exploiting the Journaling feature of the HFS+ then comparing and analyzing catalog file and journal file of the HFS+ file system. Using this deleted file's information, metadata location, and timestamp can be retrieved. By referring to journal file information, deleted files can be searched and recovered, cryptographic keys can be located, and image files can be decrypted. But it is mandatory to have a physically acquired image to perform this approach.

Carving deleted SQLite database records

Using Python script, it is possible to carve deleted records within SQLite database. The script name is "SQL parse.py" and can be downloaded from GitHub. Running strings command on the database is also helpful to recover portions of deleted entries content.

Analysis using oxygen forensic suite

Acquired data could be parsed using Oxygen Forensic Suite. Once data is parsed using Oxygen forensic suite user could search data using GUI. Below screenshot shows the information related to the acquired phone.

Figure 38. Information about the acquired phone

As shown in the screenshot above, this information includes phone name, internal name, IMEI number, platform, Acquisition type and Acquisition date.

Now, User can choose one of the two sections presented by the tool. The first section is the Common section which includes information related to the native applications. The second section includes activities related to the applications installed on IOS device.

Information like calendar events, phonebook with assigned photos, call logs, messages and voice mail could be extracted by analysis of preinstalled applications. The figure below shows call history.

Figure 39. Main Analysis Window Oxygen forensic suite

Figure 40. Call history logs

Information related to Wi-Fi access points, IP connections and locations can also be gathered using Oxygen forensic tool. For each network SSID, MAC address of the router, the timestamp of connection is presented by the tool as shown in the figure below.

Figure 41. Wi-Fi Data

The tool extracts the database, as well as plist files from the application, installed on the device. These applications are categorized into the following types.

- **Messengers:** Facebook, Skype, WhatsApp and more
- **Navigations:** Google Maps, Waze, Apple Maps and more
- **Browser:** Safari, Google Chrome and so on
- **Social networks:** Facebook, Twitter, Instagram and more
- **Travel:** Booking, Sky Scanner and more

There is advanced functionality offered by Oxygen forensic tool as listed below

- **Aggregated Contacts:** Contacts from multiple sources like Phonebook, Skype, Messages, Facebook, and more are analyzed in Aggregated Contacts. It automatically identifies common contact from multiple sources and groups them in Meta contact.
- **Dictionaries:** Words entered in messages, notes, and calendars are shown in this section
- **Links and Stats:** This section provides a tool to determine the social connection between users of mobile devices by analyzing calls, texts, multimedia, and email messages.
- **Timeline:** Calls, messages, geo data, calendar events are organized chronologically by this section. Conversation history can be followed without switching between different sections.
- **Social Graph:** Connection between mobile owner and their contacts, common contacts between multiple device owners can be detected using this tool.

This tool also offers to navigate to the file system to view all file types. SQLite database and plist file can be explored using embedded tools. Allows performing keyword search and applying the filter. The user could export finding and generate a report in a different format.

Data acquisition from iOS backup

Much valuable information can be found from the IOS backup. Users have two options to back up their data. One is using Apple iTunes software, and another is an Apple cloud storage service known as iCloud. Every time phone is connected to the iCloud or computer it creates the back up by copying files from the device. What to include in the backup can be determined by the user. Data retrieved from iCloud or iTunes may differ.

iTunes backup

Lots of information is available on a computer that has been synched with an IOS device. Historical data and passcode bypass certificates are on host computers. IOS backup file forensics involves offline back up produced by IOS devices.

In cases when logical, physical and file system acquisition is not possible then iTunes backup analysis method is very useful. Back up of iTunes is created by examiners and analyze it using forensic software.

An iPhone backup is created using free utility available for MAC and Windows platforms. It uses proprietor protocol to copy data from an IOS device to a computer. Using a cable or Wi-Fi iPhone can be synced with a computer. There is an option to create an encrypted backup, but by default, it creates an unencrypted backup. Addition access to the data stored in IOS can be accessed when the encrypted backup is cracked.

To search for the information either we could create a fresh back up, or we could extract data from the existing IOS backup file. Users create data for the prevention whether if phone damaged or lost. For example, if the device gets wiped then backup still exists. To uncover artifacts examiner needs to forensically analyze each backup.

The synchronization process gets automatically initiated once the IOS device is connected to the computer.

To disable auto-syncing in iTunes, perform the following steps:

- Navigate to iTunes | Preferences | Devices.
- Check Prevent iPods, iPhones, and iPads from syncing automatically and click OK button.
- Once synchronized settings are verified, connect the IOS device to the computer using a USB cable. iTunes automatically recognize the device if the phone is not protected with a passcode.
- iTunes will prompt the user to unlock the device if iPhone is protected with a passcode. Once iPhone is successfully synced with a computer, Backup can be performed without unlocking the phone.
- Once passcode has been supplied, and the phone is unlocked, the user is asked to enable Trust between computer and iPhone. As soon as iTunes recognizes the device, clicking on the iPhone’s icon can display a summary like iPhone’s

name, capacity, firmware version, serial number. Free space and phone number.

Pairing Records

Sets of pairing records are exchanged between the IOS device and computer when iTunes detects the iOS device. Using pairing mechanism computer establishes trusted relation with iPhone. Personal information on iDevice or backup can be initiated once the computer is paired. Starting from IOS 7 pairing mechanism has been introduced.

/var/root/Library/Lockdown/pair_records/ contains pairing records. Multiple pairing records are contained if the device is paired with multiple computers. These records are stored as property list (.plist) files. HOST ID, root certificate, device certificate and host certificate are contained within plist files. Such file has been shown below in the figure. The file shows the content within pairing record plist file.

Figure. Pairing records on iDevice

Pairing records are known as a Lockdown certificate on the computer. Preconfigured location of the pairing records is stored at below location

Windows – %AllUserProfile%AppleLockdown

Mac OS X – /private/var/db/lockdown/

host private key, Root certificate, host certificate, Escrow keybag are contained within Pairing records on computers.

Figure 42. Pairing records on computer

iTunes can back up and sync with the iPhone even in a locked state using Escrow keybag. This is a copy of class keys and a system bag that is used for the encryption in IOS. Access to all classes of data on the device without entering the password can be achieved by keybag.

Newly generated key computed from the key 0x835 is protected with Escrow keybag and saved in the escrow record on the phone. It is also plist file and located at /private/var/root/Library/Lockdown/escrow_records/

These records are protected with UntilFirstUserAuthentication protection class. It further encrypts to user’s passcode. Hence, the device passcode needs to be entered during the first time syncing the phone to iTunes.

Unencrypted backup

Commercial tools like MSAB, XRY, and Cellebrite use iTunes to create a backup for examination. Fresh installation of iTunes could also be used to create a Backup file for examination.

Follow below steps to create Unencrypted Back up,

- Using Appropriate iPhone cable connect iPhone or iOS device to the Forensic workstation.
- Launch iTunes in Forensic workstation
- Click on the iPhone icon, it would display summary page.
- In summary, page selects Back up to this computer and click Backup now.

Figure 43. Back up process

- iTunes would prompt next that sure about not encrypting Back up.
- Select Don't encrypt option to generated unencrypted Backup.
- iPhone Backup Extractor

This is a free tool for MAC OS X, which can be downloaded from the link below:

https://www.iPhonebackupextractor.com/

It extracts backup files to be located in:

~/Library/Application Support/Mobile sync/Backup/. This tool allows storing additional backup to another location like external drives.

Follow the steps below to extract Backup.

- Launch application and select option to add Backup if the backup is not detected automatically from the default location. The tool would list the available Backups on computer or iCloud. Choose the Backup need to extract.

Figure 44. Backup file

- IOS file system Back up and Individual applications can be extracted using this tool as shown in the screenshot below.

Figure 45. Applications available to extract

- Choose the target directory and extract.
- Decrypting Keychain

Except for Keychain, all the Backup files are stored unencrypted in Unencrypted Backups. To decrypt the keychain key, 0x835 needs to be extracted using a Bruteforce method using a script, or we can use Elcomsoft Phone Breaker’s Explore Keychain feature if the passcode of the user is known.

Figure 46. Extracting keychain from unencrypted Backup

The user would be asked to enter device passcode as shown in below screenshot.

Figure 47. Device passcode to explore keychain

Encrypted backup

iPhone allows extracting encrypted Backup. To protect evidence examiner could create encrypted Back up or get the access of the data which is only available with Encrypted Backup.

Following steps are performed to create an Encrypted Backup,

- Connect IOS device to the computer using Apple cable
- Launch iTunes and click on the iPhone icon which displays the connected iPhone summary.
- In summary page select location this computer and choose an encrypted backup option.

Figure 48. Set password

- set a password and encrypted Back up would be created.

This password is stored in the device itself and saved in keychain file.

Extracting encrypted backups

AES 256 algorithm in the CBC mode is used to encrypt backup files with a unique key and null initialization vector. These file keys are protected in backup Keybag with a set of class keys. Password set in iTunes through 10000 iterations of password based key derivation function 2 is protected with class keys in keybag. If the password is known then open source and commercial tools, support backup file parsing. Some tools offer to crack the password.

Elcomsoft Phone Breaker

The encrypted backup file can be cracked using this tool if back up a password is not available. It allows launching Brute-force attack on encrypted backup. Time taken to crack a password depends on the complexity of the password.

The following steps need to be performed to brute force backup a password.

- Launch Elcomsoft Phone Breaker tool and navigate to password recovery wizard in the main window. Choose source as an iOS Backup device and select manifest. plist file by navigating to the Backup file needs to be cracked.

Figure 49. Choose source manifest. plist file

- Select Attack type from the Attacks section and click start. Upon a successful attack, it would display password on the screen.

Figure 50. Attack Type

iCloud backup

Apple provides cloud computing service as iCloud storage. Data like documents, bookmarks, calendars, contacts, photos, reminders, applications and more can be kept using the iCloud account in cloud storage. Automatically and wirelessly users can backup their iOS devices to iCloud. Other services like Find My Friends and Find My iPhone also comes with iCloud.

5GB free storage is available with the sign-up. For more storage upgrade plan can be purchased. IOS 5 and later offers to the user to back up device settings and data to iCloud. Photos, Videos, documents, application data, device settings, messages, calendar and more can be found from the Backed-up data. iCloud back up can be turned on by navigating settings – iCloud. Data on the Phone can be backed up automatically when the phone is plugged in, locked, and connected to Wi-Fi. As long as space is available to create current Backup, iCloud provides a real-time copy of data available on the phone.

iTunes has also option to initiate iCloud Back up. It is incremental Backup. Data transmitted over the internet are encrypted, and stored data are in encrypted format on the server, for authentication uses secure token. Using secure token need

Figure 51. iCloud settings on Device

Extracting iCloud Backup using password

Apple’s UserID and password must be known to extract backup from iCloud. Using Apple ID and password, a user can log on to the iCloud website and can access contacts, e-mail, calendar, photos, reminders and so on. Using Elcomsoft Phone Breaker, one can extract iCloud backup. One can also download selected files from the iCloud.

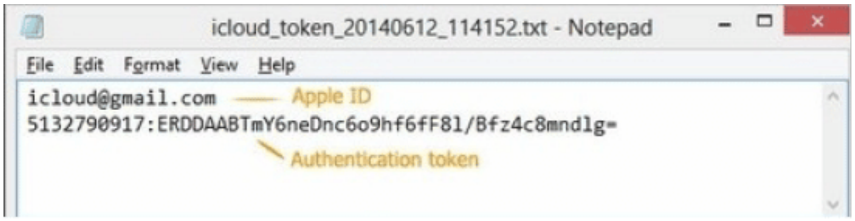
The following steps needs to be performed to extract iCloud backup.

- Launch software Elcomsoft Phone Breaker and select Download backup from iCloud from the path Tools – Apple.
- Now sign in with Apple ID and password.
- Upon successful signing, Available device backups would be listed that could be downloaded.
- Select the Backup needed and select download. Select the destination directory where you want to save when prompted. Tools would extract the file by restoring the original file names. The option is also available to restore files without original file names.

Back up Keybag in iCloud backups contains Encrypted Keychain file contents with a set of class keys. Key (0x835) protects Backup keybag which is derived from the iPhone hardware key (UID key).

Extracting iCloud Backup using Authentication Token

If one does not have username and password of the Apple ID, iCloud back up can be extracted using the Binary token available on the computer which was synced to iCloud. Using Authentication token user can bypass login of Apple iCloud as well as bypass two -factor authentication if set by the user.


icloud_token_20140612_114152.txt - Notepad
File Edit Format View Help
Apple ID
Authentication token

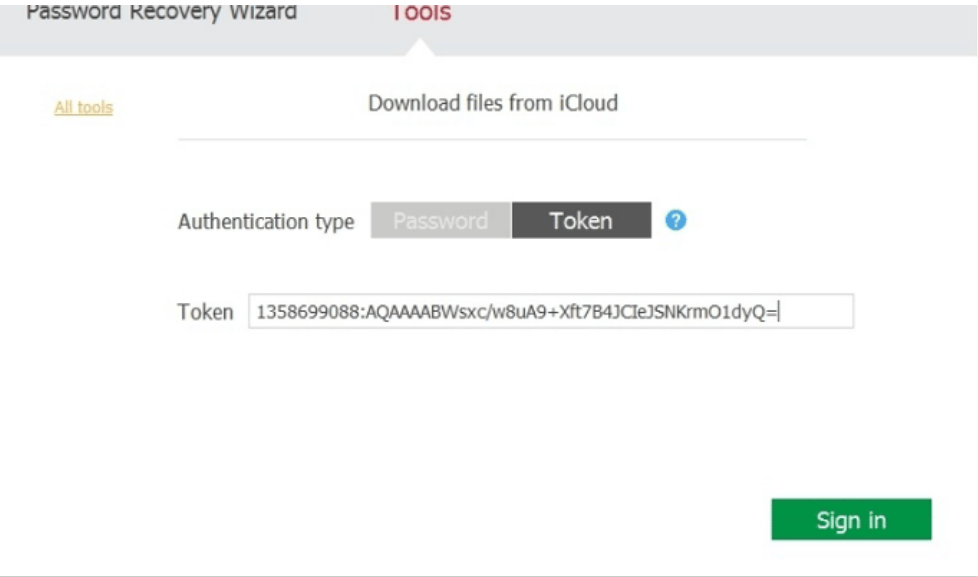
```
icloud@gmail.com
5132790917:ERDDAABTmY6neDnc6o9hf6fF81/Bfz4c8mndlg=
```

Figure 52. Authentication Token

The authentication token is a file generated so that user need not log in each time syncs with iCloud. The token is available to extract from the computer if iCloud for the Window is installed and the user has logged for the first time to sync. The token is invalid if the user logs out of the iCloud control panel.

For convenience, the user prefers to stay logged in so passwords, contacts and other types of data can be seamlessly synced. So, the probability is very high to get Authentication token from the system on which iCloud control panel is installed.

Below a screenshot from the Elcomsoft Phone Breaker shows that user has to supply authentication token instead of ID and password. It is necessary that entire authentication token string needs to be copied.



Authentication type: Token

Token: 1358699088:AQAAAABWsxc/w8uA9+Xft7B4JCIeJSNKrmO1dyQ=

Figure 53. Authentication Token applied on Tool

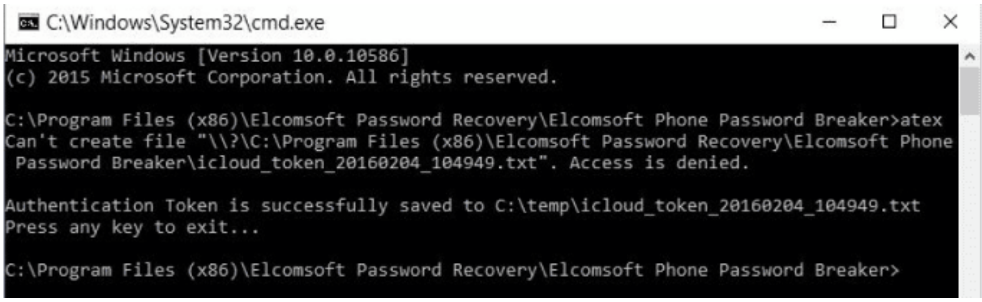
Extracting Authentication Token from the system

Elcomsoft phone breaker provides two different methods to extract tokens. Using command line tool (atex.exe) or user interface.

All the users including domain users can extract authentication token using this tool. From user’s hard drive or forensic disk image also authentication token can be extracted.

The following steps represent how to extract authentication tokens for windows users on Windows PC.

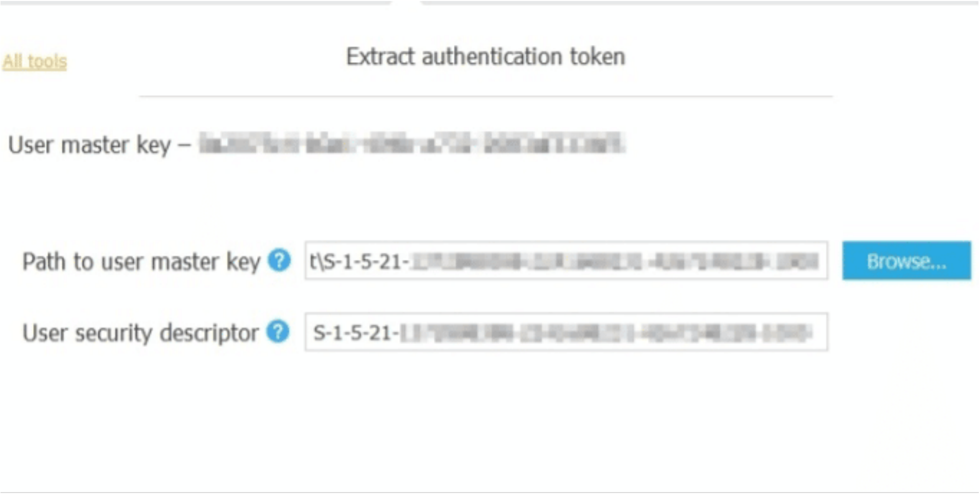
- Launch command line tool atex.exe. icloud_token_<timestamp>.txt file would be created in the same directory from which command line tool was launched. Figure. Shows the full path.
- This txt file contains authentication token and Apple ID of the iCloud control panel.
- If needs to extract authentication token for another windows user or have an image then token can be extracted using Interface on Elcomsoft phone breaker – token extraction wizard as shown in the figure.
- The path to the authentication token needs to be specified. Generally, it is %appdata%Apple computer preferences



```
Microsoft Windows [Version 10.0.10586]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Program Files (x86)\Elcomsoft Password Recovery\Elcomsoft Phone Password Breaker>atex
Can't create file "\\?\C:\Program Files (x86)\Elcomsoft Password Recovery\Elcomsoft Phone
 Password Breaker\icloud_token_20160204_104949.txt". Access is denied.

Authentication Token is successfully saved to C:\temp\icloud_token_20160204_104949.txt
Press any key to exit...

C:\Program Files (x86)\Elcomsoft Password Recovery\Elcomsoft Phone Password Breaker>
```

Figure 54. atex.exe

- Specify the path to the master key which is used to decrypt authentication token as shown in the screenshot below.



Extract authentication token

User master key – [redacted]

Path to user master key t\S-1-5-21-[redacted]

User security descriptor S-1-5-21-[redacted]

Figure 55. Master Key

The tool will now extract, decrypt and display token. This token can be exported to a file as well, and the user can log in to iCloud and download backup from iCloud using this token.

Summary

IOS devices are very popular these days, and hence there is very high chance during the investigation examiners frequently encounter iOS devices to forensically examine. New and new features are added to in IOS devices, and the vulnerable/weak feature is removed from the advanced version. With each upgraded IOS we would find new features, so we have to be updated to the technology and versions.

In this document, we covered forensic techniques for IOS device so examiners can handle investigation efficiently and could gather as much as available artifacts with efficient way. We covered types of IOS devices; IOS file system and IOS operating system. Acquisition methods for Logical extraction, file system extraction, and physical extraction was presented with step by step procedures using forensic tools. How to parse acquired data and identified artifacts and its locations covered in brief. The last section covered data extraction and analysis from the iTunes and iCloud. IOS Backup contains photos, videos, contacts, email, call logs and many important artifacts.

Posted: September 7, 2021

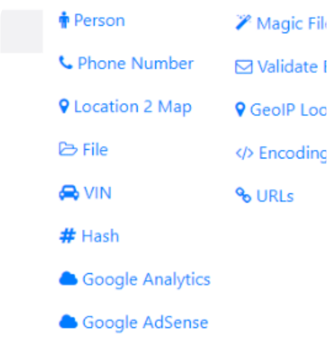
Share:

Hashim Shaikh

VIEW PROFILE

Hashim Shaikh currently works with Aujas Networks. Possessing a both OSCP and CEH, he likes exploring Kali Linux. Interests include offensive security, exploitation, privilege escalation and learning new things. His blog can be found here: http://justpentest.blogspot.in and his LinkedIn Profile here: https://in.linkedin.com/in/hashim-shaikh-oscp-45b90a48

Related Articles

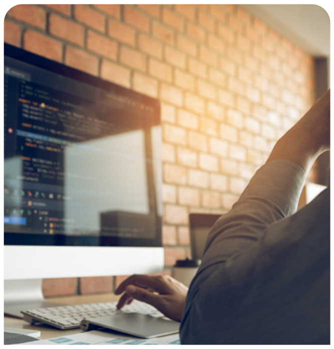


Digital forensics

Top 7 tools for intelligence-gathering purposes

July 14, 2022

Pedro Tavares

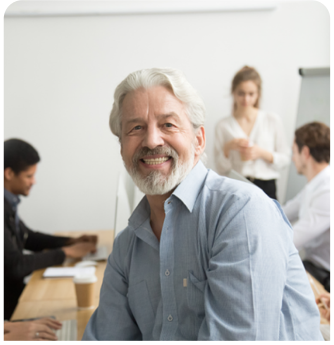


Digital forensics

Kali Linux: Top 5 tools for digital forensics

July 28, 2021

Graeme Messina



Digital forensics

Snort demo: Finding SolarWinds Sunburst indicators of compromise

July 6, 2021

Howard Poston

Digital forensics

Memory forensics demo: SolarWinds breach and Sunburst malware

June 28, 2021

Howard Poston

Topics

Hacking
Penetration testing
Cyber ranges
Capture the flag
Malware analysis
Professional development
General security
News
Security awareness
Phishing
Management, compliance & auditing
Digital forensics
Threat intelligence
DoD 8570
View all topics

Certifications

CISSP
CCSP
CGEIT
CEH
CCNA
CISA
CISM
CRISC
A+
Network+
Security+
CASP+
PMP
CySA+
CMMC
Microsoft Azure
View all certifications

Careers

IT auditor
Cybersecurity architect
Cybercrime investigator
Penetration tester
Cybersecurity consultant
Cybersecurity analyst
Cybersecurity engineer
Cybersecurity manager
Incident responder
Information security auditor
Information security manager
View all careers

Company

Contact us
About Infosec
Work at Infosec
Newsroom
Partner program

Trademarks Privacy